

IN THE CLAIMS:

1. (Currently amended) A method for asynchronous execution within a program, comprising:
 - executing code in a first thread;
 - during the executing of the code, determining whether a first keyword exists in the code, the first keyword being a flag indicating that a subsequent code element following the first keyword that may be executed out of order; and
 - executing the code element in a second thread.
2. (Original) The method of claim 1, wherein the code element is one of an instruction, a block, and a method.
3. (Currently amended) The method of claim 1, wherein the first keyword exists is usable in both a an internal definition of a method and a type definition for the method.
4. (Original) The method of claim 1, wherein the first thread is executed on a first processor and the second thread is executed on a second processor.
5. (Currently amended) ~~The method of claim 1, further comprising:~~ A method for asynchronous execution within a program, comprising:
 - executing code in a first thread;
 - determining whether a first keyword exists in the code, the first keyword indicating a code element that may be executed out of order;
 - executing the code element in a second thread;
 - determining whether a second keyword exists in the code, the second keyword indicating that execution of the code element in the second thread must complete before ~~the~~ a next code element immediately following the second keyword is executed; and
 - executing the next code element in the first thread after execution of the code element in the second thread completes.

6. (Currently amended) The method of claim 1, further comprising:
determining whether a third keyword exists in the code element, the third keyword indicating a statement that may be executed out of order; and
executing the statement in a third thread.
7. (Original) The method of claim 1, wherein the method is executed by an interpreter.
8. (Original) The method of claim 7, wherein the interpreter is a Java virtual machine.
9. (Original) The method of claim 1, wherein the second thread is a light weight thread.
10. (Currently amended) An apparatus for asynchronous execution within a program, comprising:
first execution means for executing, by the apparatus, code in a first thread;
determination means for determining, by the apparatus, whether a first keyword exists in the code, the first keyword being a type definition indicating that a subsequent code element following the first keyword that may be executed out of order; and
second execution means for executing, by the apparatus, the code element in a second thread.
11. (Original) The apparatus of claim 10, wherein the code element is one of an instruction, a block, and a method.
12. (Currently amended) The apparatus of claim 10, wherein the first keyword exists is usable in both a an internal definition of a method and a type definition for the method.
13. (Original) The apparatus of claim 10, wherein the first thread is executed on a first processor and the second thread is executed on a second processor.

14. (Currently amended) ~~The apparatus of claim 10, further comprising:~~ An apparatus for asynchronous execution within a program, comprising:

first execution means for executing, by the apparatus, code in a first thread;

determination means for determining, by the apparatus, whether a first keyword exists in the code, the first keyword indicating a code element that may be executed out of order;

second execution means for executing, by the apparatus, the code element in a second thread;

means for determining, by the apparatus, whether a second keyword exists in the code, the second keyword indicating that execution of the code element in the second thread must complete before the a next code element immediately following the second keyword is executed; and

means for executing, by the apparatus, the next code element in the first thread after execution of the code element in the second thread completes.

15. (Currently amended) The apparatus of claim ~~10~~ 14, further comprising:

means for determining, by the apparatus, whether a third keyword exists in the code element, the third keyword indicating a statement that may be executed out of order; and

means for executing, by the apparatus, the statement in a third thread.

16. (Original) The apparatus of claim 10, wherein the second thread is a light weight thread.

17. (Original) An apparatus for asynchronous execution within a program, comprising:

an interpreter; and

a program, the program including a first keyword indicating a code element that may be executed out of order,

wherein the interpreter, upon detecting the keyword, creates a light weight thread and executes the code element in the light weight thread.

18. (Original) The apparatus of claim 17, wherein the interpreter is a Java virtual machine.
19. (Currently amended) A computer program product, in a computer readable medium, for asynchronous execution within a program, comprising:
instructions for executing code in a first thread;
instructions for determining, during the executing of the code, whether a first keyword exists in the code, the first keyword being a flag indicating that a subsequent code element following the first keyword that may be executed out of order; and
instructions for executing the code element in a second thread.
20. (Original) The computer program product of claim 19, wherein the first thread is executed on a first processor and the second thread is executed on a second processor.
21. (Currently amended) ~~The computer program product of claim 19, further comprising:~~ A computer program product, in a computer readable medium, for asynchronous execution within a program, comprising:
instructions for executing code in a first thread;
instructions for determining whether a first keyword exists in the code, the first keyword indicating a code element that may be executed out of order;
instructions for executing the code element in a second thread;
instructions for determining whether a second keyword exists in the code, the second keyword indicating that execution of the code element in the second thread must complete before ~~the~~ a next code element immediately following the second keyword is executed; and
instructions for executing the next code element in the first thread after execution of the code element in the second thread completes.

22. (Currently amended) The computer program product of claim ~~19~~ 21, further comprising:

instructions for determining whether a third keyword exists in the code element,
the third keyword indicating a statement that may be executed out of order; and
instructions for executing the statement in a third thread.